This listing of claims replaces all prior versions, and listings of claims in the instant application:

Listing of Claims:

1. (Cancelled)

2. (Cancelled)

3. (Cancelled)

4. (Currently Amended) A method comprising:
    creating a first object using a runtime library wherein said first object is associated with ~~generating~~ a binary specification object for a first execution environment and further wherein said binary specification object for said first execution environment includes information on each proxy running in said first execution environment wherein a proxy comprises a wrapped interface;
    creating a second object using a runtime library wherein said second object is associated with ~~generating~~ a binary specification object for a second execution environment and further wherein said binary specification object for said second execution environment includes information on each proxy running in said second execution environment;
    issuing a call for searching for a shared library wherein said shared library is activated as ~~generating~~ a bridge object wherein said bridge object is used in mapping objects from said second execution environment to said first execution environment; and
    using said bridge object to generate a proxy wrapping an interface in said second execution environment.

5.   (Cancelled)

6.   (Currently Amended)   A method for using functionality in a second execution environment in a first execution environment comprising:

       calling a method in a proxy interface in said first execution environment <u>wherein said proxy interface is included in a bridge comprising a shared library and said first execution environment includes a binary specification having a common identity structure</u>; and

       converting said method call by said proxy interface to a corresponding method call for execution in said second execution environment, wherein said converting said method call further comprises:

              using a type description to convert parameters from said first execution environment to said second execution environment.

7.   (Cancelled)

8.   (Cancelled)

9.   (Cancelled)

10.   (Cancelled)

11.   (Original)   The method of Claim 6 wherein said second execution environment is a C++ programming language execution environment.

12.   (Currently Amended)   A method for using functionality in a second execution environment in a first execution environment comprising:

calling a method in a proxy interface in said first execution environment <u>wherein said proxy interface is included in a bridge comprising a shared library and said first execution environment includes a binary specification having a common identity structure</u>;

converting said method call by said proxy interface to a corresponding method call for execution in said second execution environment, wherein said converting said method call comprises:

using a type description to convert parameters from said first execution environment to said second execution environment; and

dispatching said corresponding method call for execution in said second execution environment to said second execution environment by said proxy interface.

13. (Original) The method of Claim 12 further comprising:

executing said corresponding method call in said second execution environment, and returning results of said execution to said proxy interface.

14. (Original) The method of Claim 13 further comprising:

using a type description to convert said returned results from said second execution environment to said first execution environment.

15. (Cancelled)

16. (Cancelled)

17. (Cancelled)

18.   (Currently Amended) A computer program product comprising computer program code for a method for using functionality in a second execution environment in a first execution environment, said method comprising:

calling a method in a proxy interface in said first execution environment <u>wherein said proxy interface is included in a bridge comprising a shared library and said first execution environment includes a binary specification having a common identity structure</u>; and

converting said method call by said proxy interface to a corresponding method call for execution in said second execution environment wherein said converting said method call further comprises:

using a type description to convert parameters from said first execution environment to said second execution environment.

19.   (Original) The computer program product of Claim 18 wherein said method further comprises:

dispatching said corresponding method call for execution in said second execution environment to said second execution environment by said proxy interface.

20.   (Cancelled)

21.   (Original) The computer program product of Claim 19 wherein said method further comprises:

executing said corresponding method call in said second execution environment, and returning results of said execution to said proxy interface.

22.   (Original) The computer program product of Claim 21 wherein said method further comprises:

using a type description to convert said returned
results from said second execution environment to said
first execution environment.

23.   (Cancelled)

24.   (Cancelled)

25.   (Cancelled)

26.   (Cancelled)

27.   (Cancelled)

28.   (Cancelled)